receiving an inbound packet at the inbound port;

classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

storing the inbound packet in the selected one of the plurality of inbound queues; and

determining when one of the plurality of inbound queues storing a plurality of packets is ready to be moved to an entry in an outbound queue associated with the outbound port, the outbound queue being capable of simultaneously storing a reference to a multiplicity of inbound queues such that a reference to each of the multiplicity of inbound queues is separately stored in a different one of a plurality of entries in the outbound queue, each of the multiplicity of inbound queues storing a plurality of packets to be separately transmitted.

---

2. (Once Amended) The method as recited in claim 1, further including:

asserting an interrupt when it is determined that one of the plurality of inbound queues is ready to be moved to an outbound queue.

3. The method as recited in claim 1, wherein classifying the inbound packet includes:

selecting inbound packet sorting criteria;

obtaining packet sorting data for the inbound packet, the packet sorting data being associated with the packet sorting criteria; and

sorting the inbound packet into one of the plurality of inbound queues according to the packet sorting data.

---

4. (Once Amended) The method as recited in claim 1, the selected one of the plurality of inbound queues corresponding to one of a plurality of outbound queues, the method further comprising:

transferring the selected one of the plurality of inbound queues storing a plurality of packets to the outbound queue associated with the outbound port such that a reference to the selected inbound queue storing a plurality of packets is stored in a single one of a plurality of entries in the outbound queue.

5. The method as recited in claim 1, wherein storing the inbound packet includes:

obtaining an available packet buffer from a free pool of available packet buffers;

placing the inbound packet in the packet buffer; and

storing the packet buffer in the inbound queue.

6. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

determining whether a number of packets in one of the plurality of inbound queues exceeds a maximum number of packets.

7. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

determining whether a number of bytes in one of the plurality of inbound queues exceeds a maximum number of bytes.

8. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue further includes:

determining whether a free pool of available memory has been depleted.

9. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue further includes:

determining whether a maximum time limit has been exceeded.

10. (Three Times Amended) A method for providing an outbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the outbound controller being adapted for forwarding packets at the outbound port, the method comprising:

providing an outbound queue associated with the outbound port and being capable of simultaneously storing a plurality of inbound queues;

receiving a notification to handle an inbound queue, the inbound queue storing a plurality of packets that are to be separately transmitted; [and]

transferring the inbound queue storing a plurality of packets to a single entry in the outbound queue associated with the outbound port such that a reference to the inbound queue storing a plurality of packets is stored in one of a plurality of entries in the outbound queue; and

repeating the receiving and transferring steps for the plurality of inbound queues such that a reference to each of the plurality of inbound queues is separately stored in a different one of the plurality of entries in the outbound queue.

11. The method as recited in claim 10, wherein receiving the notification includes:

receiving a notification from the CPU to handle the inbound queue.

12. The method as recited in claim 10, further including:

transmitting packets stored in the outbound queue.

13. The method as recited in claim 10, wherein transmitting packets includes:

selectively discarding packets stored in the outbound queue.

14. The method as recited in claim 10, wherein transmitting packets stored in the outbound queue further includes:

obtaining a next one of the plurality of inbound queues stored in the outbound queue;

transmitting selected packets stored in the next one of the plurality of inbound queues; and

releasing memory associated with the next one of the plurality of inbound queues.

15. The method as recited in claim 14, wherein releasing the memory includes:

storing the released memory in a free pool of available packet buffers.

16. The method as recited in claim 14, wherein releasing the memory includes:

forming a new inbound queue to be used by an inbound controller.

17. The method as recited in claim 14, wherein releasing the memory includes:

forming a queue to be used by the outbound controller during bi-directional operation.

---

amended
D4

18. (Once Amended) The method as recited in claim 10, wherein transferring the inbound queue to the outbound queue further includes:

ascertaining a priority of the inbound queue; and

transferring the inbound queue to a single entry in the outbound queue according to the priority of the inbound queue.

19. (Twice) A method for forwarding a packet in a router, the router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, the method comprising:

providing a plurality of inbound queues for one of the plurality of inbound ports;

providing a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of simultaneously storing a plurality of inbound queues;

receiving an inbound packet at the one of the plurality of inbound ports;

classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

D4

storing the inbound packet in the selected one of the plurality of inbound queues;

repeating the steps of receiving, [providing,] classifying, and storing until an interrupt is asserted; and

transferring one of the plurality of inbound queues storing a plurality of packets to one of the plurality of outbound queues corresponding to the packet sorting criteria when the interrupt is asserted such that a reference to the one of the plurality of inbound queues storing a plurality of packets is stored in a single one of a plurality of entries in the one of the plurality of outbound queues, wherein each of the plurality of packets in the one of the plurality of inbound queues is to be separately transmitted.

20. (Three Times Amended) An inbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the inbound controller being adapted for receiving an inbound packet at the inbound port, comprising:

a packet receiving module coupled to the inbound port, the packet receiving module being adapted for receiving an inbound packet;

wherein the memory has stored therein:

a plurality of inbound queues for the inbound port;

a classifier adapted for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

a packet storing module coupled to the classifier, the packet storing module being adapted for storing the inbound packet in the selected one of the plurality of inbound queues; and

a module adapted for determining when one of the plurality of inbound queues is ready to be moved to an entry in an outbound queue associated with the outbound port, the outbound queue being capable of simultaneously storing a multiplicity of inbound queues, a reference to each of the multiplicity of inbound queues being stored in a different one of a plurality of entries in the outbound queue [and], each of the multiplicity of inbound queues storing a plurality of packets that are to be separately transmitted.

---

21. The inbound controller as recited in claim 20, further including:

a module adapted for providing the determined one of the plurality of inbound queues.

22. The inbound controller as recited in claim 20, further including:

a module adapted for asserting an interrupt when it is determined that one of the plurality of inbound queues is ready to be moved by the CPU to the outbound queue.

23. The inbound controller as recited in claim 20, wherein the packet storing module includes:

a memory obtaining module adapted for obtaining an available packet buffer from a free pool of available packet buffers;

a module adapted for placing the inbound packet in the packet buffer; and

a module adapted for storing the packet buffer in the inbound queue.

24. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a number of packets in one of the plurality of inbound queues exceeds a maximum number of packets.

25. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a number of bytes in one of the plurality of inbound queues exceeds a maximum number of bytes.

26. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a free pool of available memory has been depleted.

27. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a maximum time limit has been exceeded.

---

28. (Three Times Amended) An outbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the outbound controller being adapted for forwarding packets at the outbound port, comprising:

a module adapted for receiving a notification to handle an inbound queue associated with the inbound port, the inbound queue storing a plurality of packets;

wherein at least one of the CPU and the memory are adapted for storing an outbound queue associated with the outbound port, the outbound queue being capable of simultaneously storing a plurality of inbound queues in a plurality of entries in the outbound queue, each of the plurality of inbound queues storing a plurality of packets that are to be separately transmitted; and

a queue transferring module adapted for transferring the inbound queue storing a plurality of packets to [an] a single entry in the outbound queue such that a reference to the inbound queue is stored in the entry in the outbound queue.

29. The outbound controller as recited in claim 28, wherein the module adapted for receiving the notification includes a module adapted for receiving the notification from the CPU.

30. The outbound controller as recited in claim 28, further including:

a module adapted for transmitting packets stored in the outbound queue.

31. The outbound controller as recited in claim 30, wherein the module adapted for transmitting packets includes:

a module adapted for selectively discarding packets stored in the outbound queue.

32. The outbound controller as recited in claim 30, wherein the module adapted for transmitting packets stored in the outbound queue includes:

a module adapted for obtaining a next one of the plurality of inbound queues stored in the outbound queue;

a packet transmission module adapted for transmitting selected packets stored in the next one of the plurality of inbound queues; and

a memory releasing module adapted for releasing memory associated with the next one of the plurality of inbound queues.

33. The outbound controller as recited in claim 32, wherein the memory releasing module includes:

a module adapted for storing the released memory in a free pool of available packet buffers.

34. The outbound controller as recited in claim 32, wherein the released memory forms a new inbound queue to be used by an inbound controller.

35. The outbound controller as recited in claim 32, wherein the released memory forms a queue to be used by the outbound controller during bi-directional operation.

---

36. (Once Amended) The outbound controller as recited in claim 28, wherein the queue transferring module is adapted for transferring the inbound queue to a single entry in the outbound queue according to a priority of the inbound queue.

37. (Twice Amended) A router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, comprising:

an inbound controller coupled to one of the plurality of inbound ports, the inbound controller being adapted for receiving an inbound packet;

wherein the memory has stored therein:

a plurality of inbound queues for the one of the plurality of inbound ports, each one of the plurality of inbound queues being capable of storing a plurality of packets that are to be separately transmitted;

a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of simultaneously storing a plurality of inbound queues such that a reference to each of the plurality of inbound queues is stored in a different [in] one of a plurality of entries in the one of the plurality of outbound queues; and

a classifier coupled to the inbound controller, the classifier being adapted for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria, the selected one of the plurality of inbound queues being associated with one of the plurality of outbound queues;

wherein the inbound controller is adapted for storing the inbound packet in the selected one of the plurality of inbound queues.

D6

38. (Once Amended) The router as recited in claim 37, further including:

an outbound controller coupled to the inbound controller;

wherein the inbound controller selects one of the plurality of inbound queues to be transferred to the outbound controller;

wherein the outbound controller is adapted for storing a reference to the selected one of the plurality of inbound queues in an entry in one of the plurality of outbound queues associated with the packet sorting criteria and transmitting packets stored in the one of the plurality of outbound queues.

39. The router as recited in claim 37, wherein the inbound controller further includes:

a memory obtaining module coupled to the classifier, the memory obtaining module being adapted for obtaining memory for an inbound packet to permit the inbound packet to be stored in the selected one of the plurality of inbound queues in which the inbound packet is classified.

40. The router as recited in claim 38, wherein the outbound controller further includes:

a memory releasing module adapted for releasing selected packet buffers associated with packets stored in the one of the plurality of outbound queues.

41. The router as recited in claim 40, wherein the memory further includes a free pool of available packet buffers and the memory releasing module is adapted for releasing the selected packet buffers into the free pool.

42. The router as recited in claim 38, wherein the outbound controller further includes:

a memory releasing module adapted for providing a new inbound queue to the inbound controller to replace the selected one of the plurality of inbound queues.

---

43. (Three Times Amended) An encryption system, comprising:

an inbound controller adapted for receiving an inbound packet;

a classifier coupled to the inbound controller and adapted for classifying and storing the inbound packet in one of a plurality of inbound queues;

an outbound controller adapted for receiving the one of the plurality of inbound queues, the one of the plurality of inbound queues storing a plurality of packets to be separately transmitted; and

an encryption box coupled to the outbound controller, the encryption box being adapted for encrypting the one of the plurality of inbound queues to provide an encrypted inbound queue to the outbound controller for transmission, wherein the outbound controller includes an outbound classifier adapted for classifying the encrypted inbound queue in one of a plurality of outbound queues associated with a plurality of outbound ports, the outbound controller being adapted for storing a reference to the encrypted inbound queue in a single entry in the one of the plurality of outbound queues, and transmitting data stored in the one of the plurality of outbound queues.

---

45. (Once Amended) The method as recited in claim 2, further comprising:

when the interrupt is asserted, transferring the one of the plurality of inbound queues to an entry in the outbound queue or an outbound controller associated with the outbound queue.

46. (Once Amended) The method as recited in claim 45, wherein transferring the one of the plurality of inbound queues to an entry in the outbound queue or an outbound controller associated with the outbound queue is performed by the CPU.

D8

47. (Once Amended) The method as recited in claim 45, wherein transferring the one of the plurality of inbound queues to an entry in the outbound queue or an outbound controller associated with the outbound queue comprises:

transferring a reference to the one of the plurality of inbound queues to an entry in an outbound queue corresponding to a priority associated with the one of the plurality of inbound queues.

48. (Once Amended) The method as recited in claim 45, wherein transferring the one of the plurality of inbound queues to the outbound queue or an outbound controller associated with the outbound queue comprises:

transferring a pointer to the one of the plurality of inbound queues to an entry in an outbound queue associated with the one of the plurality of inbound queues.

---

49. The method as recited in claim 43, wherein the inbound queue stores therein a plurality of packets, and wherein the encryption box does not encrypt each of the plurality of packets.

50. The method as recited in claim 43, wherein the encryption box is adapted for encrypting the inbound queue as an entity such that a single encryption step is performed.

51. (Once Amended) The method as recited in claim 10, wherein transferring the inbound queue to a single entry in the outbound queue is performed by the CPU in response to an interrupt.

52. (Once Amended) The method as recited in claim 1, wherein the outbound queue comprises a plurality of entries, each of the plurality of entries simultaneously storing or identifying one of the multiplicity of inbound queues.

53. (Twice Amended) A computer-readable medium storing thereon computer-readable instructions for forwarding a packet in a router, the router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, the method comprising:

instructions for providing a plurality of inbound queues for one of the plurality of inbound ports;

instructions for providing a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of simultaneously storing a plurality of inbound queues such that a reference to each of the plurality of inbound queues is simultaneously stored in a different one of a plurality of entries in the one of the plurality of outbound queues;

instructions for receiving an inbound packet at the one of the plurality of inbound ports;

instructions for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

instructions for storing the inbound packet in the selected one of the plurality of inbound queues;

instructions for repeating the steps of receiving, [providing,] classifying, and storing until an interrupt is asserted; and

instructions for transferring one of the plurality of inbound queues to one of the plurality of outbound queues corresponding to the packet sorting criteria when the interrupt is asserted such

that a pointer to the transferred one of the plurality of inbound queues is stored in one of a plurality of entries in the one of the plurality of outbound queues, the one of the plurality of inbound queues storing a plurality of packets that are to be separately transmitted.

54. (Twice Amended) An apparatus for forwarding a packet in a router, the router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, the method comprising:

means for providing a plurality of inbound queues for one of the plurality of inbound ports;

D9 means for providing a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of simultaneously storing a plurality of inbound queues such that a reference to each of the plurality of inbound queues is stored in a different one of a plurality of entries in the one of the plurality of outbound queues;

means for receiving an inbound packet at the one of the plurality of inbound ports;

means for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

means for storing the inbound packet in the selected one of the plurality of inbound queues;

means for repeating the steps of receiving, [providing,] classifying, and storing until an interrupt is asserted; and

means for transferring one of the plurality of inbound queues to one of the plurality of outbound queues corresponding to the packet sorting criteria when the interrupt is asserted such that a reference to the transferred one of the plurality of inbound queues is stored in a single one of a plurality of entries in the one of the plurality of outbound queues, the one of the plurality of inbound queues storing a plurality of packets that are to be separately transmitted.

REPLACEMENT SHEETS

1. (Three Times Amended) A method for providing an inbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the inbound controller being adapted for receiving an inbound packet at the inbound port, the method comprising:

providing a plurality of inbound queues for the inbound port;

receiving an inbound packet at the inbound port;

classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

storing the inbound packet in the selected one of the plurality of inbound queues; and

determining when one of the plurality of inbound queues storing a plurality of packets is ready to be moved to an entry in an outbound queue associated with the outbound port, the outbound queue being capable of simultaneously storing a reference to a multiplicity of inbound queues such that a reference to each of the multiplicity of inbound queues is separately stored in a different one of a plurality of entries in the outbound queue, each of the multiplicity of inbound queues storing a plurality of packets to be separately transmitted.

2. (Once Amended) The method as recited in claim 1, further including:

asserting an interrupt when it is determined that one of the plurality of inbound queues is ready to be moved to an outbound queue.

3. The method as recited in claim 1, wherein classifying the inbound packet includes:

selecting inbound packet sorting criteria;

obtaining packet sorting data for the inbound packet, the packet sorting data being associated with the packet sorting criteria; and

sorting the inbound packet into one of the plurality of inbound queues according to the packet sorting data.

4. (Once Amended) The method as recited in claim 1, the selected one of the plurality of inbound queues corresponding to one of a plurality of outbound queues, the method further comprising:

transferring the selected one of the plurality of inbound queues storing a plurality of packets to the outbound queue associated with the outbound port such that a reference to the selected inbound queue storing a plurality of packets is stored in a single one of a plurality of entries in the outbound queue.

5. The method as recited in claim 1, wherein storing the inbound packet includes:

obtaining an available packet buffer from a free pool of available packet buffers;

placing the inbound packet in the packet buffer; and

storing the packet buffer in the inbound queue.

6. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

determining whether a number of packets in one of the plurality of inbound queues exceeds a maximum number of packets.

7. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

determining whether a number of bytes in one of the plurality of inbound queues exceeds a maximum number of bytes.

8. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue further includes:

determining whether a free pool of available memory has been depleted.

9. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue further includes:

determining whether a maximum time limit has been exceeded.

10. (Three Times Amended) A method for providing an outbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the outbound controller being adapted for forwarding packets at the outbound port, the method comprising:

providing an outbound queue associated with the outbound port and being capable of simultaneously storing a plurality of inbound queues;

receiving a notification to handle an inbound queue, the inbound queue storing a plurality of packets that are to be separately transmitted;

transferring the inbound queue storing a plurality of packets to a single entry in the outbound queue associated with the outbound port such that a reference to the inbound queue storing a plurality of packets is stored in one of a plurality of entries in the outbound queue; and

repeating the receiving and transferring steps for the plurality of inbound queues such that a reference to each of the plurality of inbound queues is separately stored in a different one of the plurality of entries in the outbound queue.

11. The method as recited in claim 10, wherein receiving the notification includes:

receiving a notification from the CPU to handle the inbound queue.

12. The method as recited in claim 10, further including:

transmitting packets stored in the outbound queue.

13. The method as recited in claim 10, wherein transmitting packets includes:

selectively discarding packets stored in the outbound queue.

14. The method as recited in claim 10, wherein transmitting packets stored in the outbound queue further includes:

obtaining a next one of the plurality of inbound queues stored in the outbound queue;

transmitting selected packets stored in the next one of the plurality of inbound queues; and

releasing memory associated with the next one of the plurality of inbound queues.

15. The method as recited in claim 14, wherein releasing the memory includes:

storing the released memory in a free pool of available packet buffers.

16. The method as recited in claim 14, wherein releasing the memory includes:

forming a new inbound queue to be used by an inbound controller.

17. The method as recited in claim 14, wherein releasing the memory includes:

forming a queue to be used by the outbound controller during bi-directional operation.

18. (Once Amended) The method as recited in claim 10, wherein transferring the inbound queue to the outbound queue further includes:

ascertaining a priority of the inbound queue; and

transferring the inbound queue to a single entry in the outbound queue according to the priority of the inbound queue.

19. (Twice) A method for forwarding a packet in a router, the router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, the method comprising:

providing a plurality of inbound queues for one of the plurality of inbound ports;

providing a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of simultaneously storing a plurality of inbound queues;

receiving an inbound packet at the one of the plurality of inbound ports;

classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

storing the inbound packet in the selected one of the plurality of inbound queues;

repeating the steps of receiving, classifying, and storing until an interrupt is asserted; and

transferring one of the plurality of inbound queues storing a plurality of packets to one of the plurality of outbound queues corresponding to the packet sorting criteria when the interrupt is asserted such that a reference to the one of the plurality of inbound queues storing a plurality of packets is stored in a single one of a plurality of entries in the one of the plurality of outbound queues, wherein each of the plurality of packets in the one of the plurality of inbound queues is to be separately transmitted.

20. (Three Times Amended) An inbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the inbound controller being adapted for receiving an inbound packet at the inbound port, comprising:

a packet receiving module coupled to the inbound port, the packet receiving module being adapted for receiving an inbound packet;

wherein the memory has stored therein:

a plurality of inbound queues for the inbound port;

a classifier adapted for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

a packet storing module coupled to the classifier, the packet storing module being adapted for storing the inbound packet in the selected one of the plurality of inbound queues; and

a module adapted for determining when one of the plurality of inbound queues is ready to be moved to an entry in an outbound queue associated with the outbound port, the outbound queue being capable of simultaneously storing a multiplicity of inbound queues, a reference to each of the multiplicity of inbound queues being stored in a different one of a plurality of entries in the outbound queue, each of the multiplicity of inbound queues storing a plurality of packets that are to be separately transmitted.

21. The inbound controller as recited in claim 20, further including:

a module adapted for providing the determined one of the plurality of inbound queues.

22. The inbound controller as recited in claim 20, further including:

a module adapted for asserting an interrupt when it is determined that one of the plurality of inbound queues is ready to be moved by the CPU to the outbound queue.

23. The inbound controller as recited in claim 20, wherein the packet storing module includes:

a memory obtaining module adapted for obtaining an available packet buffer from a free pool of available packet buffers;

a module adapted for placing the inbound packet in the packet buffer; and

a module adapted for storing the packet buffer in the inbound queue.

24. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a number of packets in one of the plurality of inbound queues exceeds a maximum number of packets.

25. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a number of bytes in one of the plurality of inbound queues exceeds a maximum number of bytes.

26. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a free pool of available memory has been depleted.

27. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a maximum time limit has been exceeded.

28. (Three Times Amended) An outbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the outbound controller being adapted for forwarding packets at the outbound port, comprising:

a module adapted for receiving a notification to handle an inbound queue associated with the inbound port, the inbound queue storing a plurality of packets;

wherein at least one of the CPU and the memory are adapted for storing an outbound queue associated with the outbound port, the outbound queue being capable of simultaneously storing a plurality of inbound queues in a plurality of entries in the outbound queue, each of the plurality of inbound queues storing a plurality of packets that are to be separately transmitted; and

a queue transferring module adapted for transferring the inbound queue storing a plurality of packets to a single entry in the outbound queue such that a reference to the inbound queue is stored in the entry in the outbound queue.

29. The outbound controller as recited in claim 28, wherein the module adapted for receiving the notification includes a module adapted for receiving the notification from the CPU.

30. The outbound controller as recited in claim 28, further including:

a module adapted for transmitting packets stored in the outbound queue.

31. The outbound controller as recited in claim 30, wherein the module adapted for transmitting packets includes:

a module adapted for selectively discarding packets stored in the outbound queue.

32. The outbound controller as recited in claim 30, wherein the module adapted for transmitting packets stored in the outbound queue includes:

a module adapted for obtaining a next one of the plurality of inbound queues stored in the outbound queue;

a packet transmission module adapted for transmitting selected packets stored in the next one of the plurality of inbound queues; and

a memory releasing module adapted for releasing memory associated with the next one of the plurality of inbound queues.

33. The outbound controller as recited in claim 32, wherein the memory releasing module includes:

a module adapted for storing the released memory in a free pool of available packet buffers.

34. The outbound controller as recited in claim 32, wherein the released memory forms a new inbound queue to be used by an inbound controller.

35. The outbound controller as recited in claim 32, wherein the released memory forms a queue to be used by the outbound controller during bi-directional operation.

36. (Once Amended) The outbound controller as recited in claim 28, wherein the queue transferring module is adapted for transferring the inbound queue to a single entry in the outbound queue according to a priority of the inbound queue.

37. (Twice Amended) A router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, comprising:

an inbound controller coupled to one of the plurality of inbound ports, the inbound controller being adapted for receiving an inbound packet;

wherein the memory has stored therein:

a plurality of inbound queues for the one of the plurality of inbound ports, each one of the plurality of inbound queues being capable of storing a plurality of packets that are to be separately transmitted;

a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of simultaneously storing a plurality of inbound queues such that a reference to each of the plurality of inbound queues is stored in a different one of a plurality of entries in the one of the plurality of outbound queues; and

a classifier coupled to the inbound controller, the classifier being adapted for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria, the selected one of the plurality of inbound queues being associated with one of the plurality of outbound queues;

wherein the inbound controller is adapted for storing the inbound packet in the selected one of the plurality of inbound queues.

38. (Once Amended) The router as recited in claim 37, further including:

an outbound controller coupled to the inbound controller;

wherein the inbound controller selects one of the plurality of inbound queues to be transferred to the outbound controller;

wherein the outbound controller is adapted for storing a reference to the selected one of the plurality of inbound queues in an entry in one of the plurality of outbound queues associated with the packet sorting criteria and transmitting packets stored in the one of the plurality of outbound queues.

39. The router as recited in claim 37, wherein the inbound controller further includes:

a memory obtaining module coupled to the classifier, the memory obtaining module being adapted for obtaining memory for an inbound packet to permit the inbound packet to be stored in the selected one of the plurality of inbound queues in which the inbound packet is classified.

40. The router as recited in claim 38, wherein the outbound controller further includes:

a memory releasing module adapted for releasing selected packet buffers associated with packets stored in the one of the plurality of outbound queues.

41. The router as recited in claim 40, wherein the memory further includes a free pool of available packet buffers and the memory releasing module is adapted for releasing the selected packet buffers into the free pool.

42. The router as recited in claim 38, wherein the outbound controller further includes:

a memory releasing module adapted for providing a new inbound queue to the inbound controller to replace the selected one of the plurality of inbound queues.

43. (Three Times Amended) An encryption system, comprising:

an inbound controller adapted for receiving an inbound packet;

a classifier coupled to the inbound controller and adapted for classifying and storing the inbound packet in one of a plurality of inbound queues;

an outbound controller adapted for receiving the one of the plurality of inbound queues, the one of the plurality of inbound queues storing a plurality of packets to be separately transmitted; and

an encryption box coupled to the outbound controller, the encryption box being adapted for encrypting the one of the plurality of inbound queues to provide an encrypted inbound queue to the outbound controller for transmission, wherein the outbound controller includes an outbound classifier adapted for classifying the encrypted inbound queue in one of a plurality of outbound queues associated with a plurality of outbound ports, the outbound controller being adapted for storing a reference to the encrypted inbound queue in a single entry in the one of the plurality of outbound queues, and transmitting data stored in the one of the plurality of outbound queues.

45. (Once Amended) The method as recited in claim 2, further comprising:

when the interrupt is asserted, transferring the one of the plurality of inbound queues to an entry in the outbound queue or an outbound controller associated with the outbound queue.

46. (Once Amended) The method as recited in claim 45, wherein transferring the one of the plurality of inbound queues to an entry in the outbound queue or an outbound controller associated with the outbound queue is performed by the CPU.

47. (Once Amended) The method as recited in claim 45, wherein transferring the one of the plurality of inbound queues to an entry in the outbound queue or an outbound controller associated with the outbound queue comprises:

transferring a reference to the one of the plurality of inbound queues to an entry in an outbound queue corresponding to a priority associated with the one of the plurality of inbound queues.

48. (Once Amended) The method as recited in claim 45, wherein transferring the one of the plurality of inbound queues to the outbound queue or an outbound controller associated with the outbound queue comprises:

transferring a pointer to the one of the plurality of inbound queues to an entry in an outbound queue associated with the one of the plurality of inbound queues.

49. The method as recited in claim 43, wherein the inbound queue stores therein a plurality of packets, and wherein the encryption box does not encrypt each of the plurality of packets.

50. The method as recited in claim 43, wherein the encryption box is adapted for encrypting the inbound queue as an entity such that a single encryption step is performed.

51. (Once Amended) The method as recited in claim 10, wherein transferring the inbound queue to a single entry in the outbound queue is performed by the CPU in response to an interrupt.

52. (Once Amended) The method as recited in claim 1, wherein the outbound queue comprises a plurality of entries, each of the plurality of entries simultaneously storing or identifying one of the multiplicity of inbound queues.

53. (Twice Amended) A computer-readable medium storing thereon computer-readable instructions for forwarding a packet in a router, the router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, the method comprising:

instructions for providing a plurality of inbound queues for one of the plurality of inbound ports;

instructions for providing a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of simultaneously storing a plurality of inbound queues such that a reference to each of the plurality of inbound queues is simultaneously stored in a different one of a plurality of entries in the one of the plurality of outbound queues;

instructions for receiving an inbound packet at the one of the plurality of inbound ports;

instructions for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

instructions for storing the inbound packet in the selected one of the plurality of inbound queues;

instructions for repeating the steps of receiving, classifying, and storing until an interrupt is asserted; and

instructions for transferring one of the plurality of inbound queues to one of the plurality of outbound queues corresponding to the packet sorting criteria when the interrupt is asserted such

that a pointer to the transferred one of the plurality of inbound queues is stored in one of a plurality of entries in the one of the plurality of outbound queues, the one of the plurality of inbound queues storing a plurality of packets that are to be separately transmitted.

54. (Twice Amended) An apparatus for forwarding a packet in a router, the router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, the method comprising:

means for providing a plurality of inbound queues for one of the plurality of inbound ports;

means for providing a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of simultaneously storing a plurality of inbound queues such that a reference to each of the plurality of inbound queues is stored in a different one of a plurality of entries in the one of the plurality of outbound queues;

means for receiving an inbound packet at the one of the plurality of inbound ports;

means for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

means for storing the inbound packet in the selected one of the plurality of inbound queues;

means for repeating the steps of receiving, classifying, and storing until an interrupt is asserted; and

means for transferring one of the plurality of inbound queues to one of the plurality of outbound queues corresponding to the packet sorting criteria when the interrupt is asserted such that a reference to the transferred one of the plurality of inbound queues is stored in a single one of a plurality of entries in the one of the plurality of outbound queues, the one of the plurality of inbound queues storing a plurality of packets that are to be separately transmitted.